

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representation of
The original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

BEST AVAILABLE COPY



Eur päisches Patentamt
European Patent Office
Office ur péen d s brevets



(11) Veröffentlichungsnummer: **0 645 702 A1**

(12)

EUROPÄISCHE PATENTANMELDUNG

(21) Anmeldenummer: **93115467.8**

(51) Int. Cl.⁶: **G06F 9/46, H04M 3/36,
H04Q 3/545**

(22) Anmeldetag: **24.09.93**

(43) Veröffentlichungstag der Anmeldung:
29.03.95 Patentblatt 95/13

(84) Benannte Vertragsstaaten:
AT BE CH DE DK ES FR GB IT LI NL SE

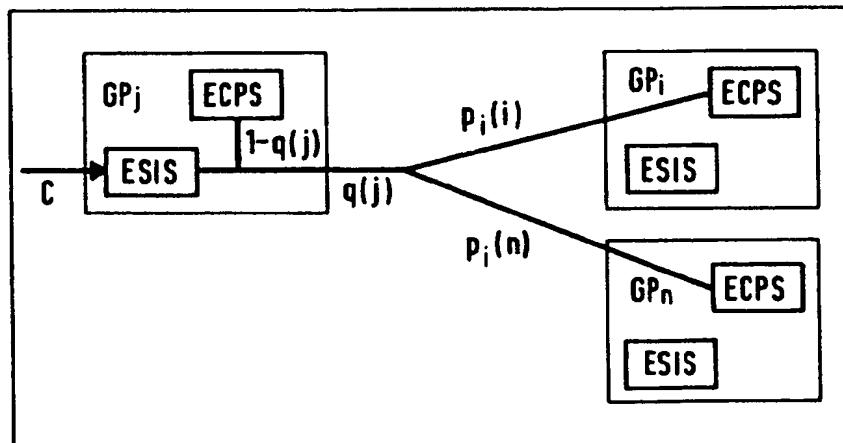
(71) Anmelder: **SIEMENS AKTIENGESELLSCHAFT
Wittelsbacherplatz 2
D-80333 München (DE)**

(72) Erfinder: **Witte, Martin, Dr.
Josephsburgstrasse 5
D-81673 München (DE)
Erfinder: Oehlerich, Jörg
Präntlweg 1
D-82131 Stockdorf (DE)
Erfinder: Held, Walter
Isardamm 129
D-82538 Geretsried (DE)**

(54) **Verfahren zum Lastausgleich in einem Multiprozessorsystem.**

(57) Verfahren zum Lastausgleich in einem Multiprozessorsystem, insbesondere ein Multiprozessorsystem eines Kommunikationssystems, bei dem anfallende Aufgaben von mehreren Prozessoren unter Realzeitbedingungen abgearbeitet werden können. Geraten aufgrund von lokalen Starklastsituationen einzelne Prozessoren in Überlast, so wird ein Abbau der lokalen Überlast dadurch erreicht, daß in dieser Lastsituation weitere eintreffende Anforderungen an den Prozessor so weit als möglich/nötig an andere, nicht überlastete Prozessoren weitergeleitet werden.

FIG 3



EP 0 645 702 A1

Die Erfindung betrifft ein Verfahren zum Lastausgleich in einem Multiprozessorsystem, bei dem anfallende Aufgaben von mehreren Prozessoren unter Realzeitbedingungen abgearbeitet werden können.

Wenn es sich bei dem Multiprozessorsystem um ein prozessorgesteuertes Kommunikationssystem handelt, wird das Kommunikationssystem aufgrund eines prognostizierten Vermittlungsaufkommens und/oder einzuhaltender Mindestanforderungen, wie beispielsweise die an die Verbindungsauflauzeiten (Through Connection Delay), dimensioniert. Ein wesentlicher Parameter der die Dimensionierung eines Vermittlungssystems bestimmt, wird mit dem zu erwartenden Verkehrswert vorgeben. Dieser Verkehrswert ergibt sich dabei aus dem Quotienten der sich aus der Summe der Belegungsdauer - bezogen auf die Belegungen in einem Beobachtungsintervall errechnet. Ein nach den herkömmlichen Verkehrserwartungen dimensioniertes prozessorgesteuertes Kommunikationssystem muß jedoch weitgehendst auch alle ankommenden Vermittlungsanfragen bei auftretenden Spitzenbelastungen, die an bestimmten Tagen, einzelnen Tagesstunden oder nicht vorherbestimmbaren Ereignissen auftreten, beispielsweise unter Zuhilfenahme von Lastausgleichsverfahren, abarbeiten können.

Der Erfindung liegt die Aufgabe zugrunde, ein Lastausgleichsverfahren für ein Realzeitsystem anzugeben, das gewährleistet, daß auch in Starklastsituationen möglichst wenig Prozessoranforderungen abgewiesen werden.

Die Aufgabe wird durch die Merkmale des Patentanspruchs 1 gelöst.

Die Erfindung bringt den Vorteil mit sich, daß auf Grund der Berücksichtigung von Lastzuständen von Prozessoren die Überlast einzelner Prozessoren auf freie, nicht überlastete Prozessoren verteilt wird, wobei die verteilten Aufgaben keine zusätzlichen Lastschwankungen innerhalb von vermaschten Prozessoren verursachen sowie die Einhaltung der Through Connection Delays dadurch nicht beeinflußt werden.

Des Weiteren bringt die Erfindung neben dem Vorteil, daß die in "Unterlast" befindlichen Prozessoren vor Überlastung durch verteilte Aufgaben geschützt werden, den weiteren Vorteil mit sich, daß die Auslastung des Prozessors an seiner "Hochlastschwelle" orientiert wird.

Eine Ausgestaltung der Erfindung ist, daß die Lastkennwerte mit einem Hysteresewert beaufschlagt werden, was den Vorteil mit sich bringt, daß Überschwingungen bei Eintritt in den Überlastfall vermieden werden.

Eine weitere Ausgestaltung der Erfindung ist, daß zusätzliche Aufgaben auch durch einen Prozessor abgearbeitet werden, dessen Lastzustand mit "high" eingestuft wird, dies bringt den Vorteil mit sich, daß die Anzahl der zugewiesenen fremden Aufgaben pro Prozessor an seiner zur Verfügung stehenden freien Kapazität orientiert werden.

Eine Ausgestaltung der Erfindung ist, daß der Anteil der von einem in Überlast befindlichen Prozessor wegzuverteilenden Aufgaben aus der Lastkennzahl ermittelt wird.

Eine Ausgestaltung der Erfindung ist, daß zur selbst bearbeiteten Last die verteilte Last geschätzt dazu addiert wird. Dies bringt den Vorteil mit sich, daß erst wenn die geschätzte tatsächlich am Prozessor anliegende Last unter eine Hochlastschwelle sinkt und eine vermittelungstechnische Überlasteinstufung auf Null gesetzt und eine Sicherheitsperiode eingehalten wurde, die Überlastsituation des Prozessors als beendet angesehen wird.

Vorteilhafte Weiterbildungen der Erfindung sind in den Unteransprüchen angegeben.
Weitere Besonderheiten der Erfindung werden aus der nachfolgenden näheren Erläuterung eines Ausführungsbeispiels, in Form eines Kommunikationssystems, anhand der Zeichnungen ersichtlich.

Es zeigen:

- Fig 1. einen Teil eines Kommunikationssystems,
- Fig 2. Prozeß-Verarbeitungseinheiten eines solchen zum Kommunikationssystem gehörenden Gruppenprozessors,
- Fig 3. ein Schema für das erfindungsgemäße Verfahren, und
- Fig 4. ein Ablaufdiagramm.

Fig 1 zeigt schematisch, in einem zum Verständnis der Erfindung erforderlichen Umfang Komponenten eines Kommunikationssystems K. Diese Komponenten sind Teilnehmerendgeräte T_{n1},...,T_{nm}, Anschlußbaugruppen LTG₁,...,LTG_n für den Anschluß der mit diesen Teilnehmerendgeräten verbindenden Teilnehmeranschlußleitungen, Gruppenprozessoren GP_n, ein Koordinationsprozessor CP oder ein weiterer die Funktionen eines Koordinationsprozessors übernehmender Gruppenprozessor GP, eine Schnittstelleneinrichtung SE zu weiteren Vermittelungssystemen, eine Serviceeinheit SU, einem im Asynchronen Transfer Mode arbeitenden Multiplexer AMX sowie ein Koppelnetz SNB, daß mit einem Asynchronem Transfer Mode ATM Verbindungswege zu gewünschten Teilnehmerendgeräten T_{nm} durchschaltet. Die Ahschlußbaugruppen LTG_n fassen jeweils eine Mehrzahl von Teilnehmerendgeräten zusammen. Vermittelungstechnische Aufgaben die in den Anschlußbaugruppen LTG_n anfallen werden jeweils durch mindestens einen Gruppenprozessor GP_i bearbeitet. Die in einem Kommunikationssystem K verwendeten Gruppenprozessoren GP_n sind

dabei im wesentlichen alle gleichartig strukturiert. Die Gruppenprozessoren GPn haben über den Multiplexer AMX Zugang zu weiterverarbeitenden Einheiten des Kommunikationssystems K. An diesen Multiplexer AMX sind neben den Gruppenprozessoren GPn die Service Einheit SU, das Koppelnetz SNB als auch weitere Prozessoren, wie beispielsweise ein Koordinationsprozessor CP angeschlossen. Über den Asynchronen Transfer Mode Multiplexer AMX besteht unter anderem auch ein Zugang zu anderen Vermittlungssystemen.

Die Gruppenprozessoren GPn sind hauptsächlich für den Verbindungsaufbau und Verbindungsabbau zuständig. Sie besitzen zudem zur Abwicklung solcher vermittlungstechnischer Aufgaben Kenndaten über die speziellen Ausprägungen der an ihnen angeschlossenen Anschlußbaugruppen LTGn. Der Datenaustausch zwischen den Gruppenprozessoren GPn findet über den AMX statt.

Fig. 2 zeigt einige für die Erfindung wesentliche Systemkomponenten des Gruppenprozessors GPi. Der Gruppenprozessor GPi wird intern in eine Vielzahl virtueller Prozessoreinheiten VCPUs unterteilt, die jeweils für verschiedene Aufgaben zuständig sind. Die Zuweisung der Rechenzeit der "Central Processor Unit" CPU an die jeweiligen Mikroprozessoreinheit VCPUs erfolgt über ein spezielles Zuweisungsverfahren (Scheduling). Im Zusammenhang mit der Erfindung ist die virtuelle Prozessoreinheit VCPU-CallIP von besonderem Interesse, da sie für die Vermittlung der Telefonverbindungen zuständig ist. Die virtuelle Prozessoreinheit VCPU-CallIP setzt sich intern aus einer Vielzahl von Prozessen zusammen:

Beispielsweise seien hier einige Prozesse sowie deren Aufgaben aufgeführt:

- ESIS: Extended Signalling and Interworking Subsystem Das EWSX Signalling and Interworking Subsystem ESIS bildet die Schnittstelle zwischen den verschiedenen an die Anschlußbaugruppen LTGn zugeordneten Teilnehmerendgeräten Tn1,...,Tnm und führt die Anpassung zwischen externen Signalisierungssystemen und dem internen Message Interface durch.
- ECPS: Extended Call Processing Subsystem Dieses Subsystem ist für die Verarbeitung des Call-Routing, Call-Setups, Call-Cleardown sowie spezielle Features zuständig. Zur Abarbeitung dieser Prozesse können spezielle Dienstleistungen, wie die des ATS(Access Transaction Segment), UTS (User Transaction Segment), AS (Associator Segment)...., in Anspruch genommen werden.

Diese Prozesse kommunizieren miteinander über sogenannte Messages. Pro Verbindung eines Calls, einer von einem Teilnehmerendgerät Tnm ausgelösten Prozedur zum Verbindungsaufbau, werden dabei eine Vielzahl von Messages zwischen den einzelnen Prozessen verschickt. Dieser im Einzelfall aus weit über 100 Meldungsprozeduren bestehende Meldungsverkehr läuft pro Call auf einen einzigen Gruppenprozessor GPi ab. Wird ein Call verteilt, laufen die Messages zwischen dem Prozeß ESIS, der sich auf dem "Heim-Gruppenprozessor" GPj befindet und den ECPS Prozessen sowie weiteren vermittlungstechnischen Prozessen über das ATM Netz auf einen noch "belastbaren" Gruppenprozessor GPx ab.

Fig. 3 zeigt schematisch ein Verteilschema, daß an den Gruppenprozessor GPj anliegende Calls C an noch "belastbare" Gruppenprozessoren GPx weiterleitet.

Ein in den Gruppenprozessor GPn implementierter Load-Manager GPLM, der durch die virtuelle Prozessoreinheit Operation & Maintainance (high priority) VCPU-OAMH jeweils eine Prozessorzeit zur Abarbeitung seiner Aufgaben zugewiesen bekommt, ermittelt den augenblicklichen Lastzustand (Gruppenprozessor Load State) GPLS des Gruppenprozessors sowie einen Wert OPL (Overload Priority Level) der eine spezifische vermittlungstechnische Überlastabwehrstufe angibt. Diese spezifische vermittlungstechnische Überlastabwehrstufe OPL wird jeweils im Lastzustand - "overload"- des Gruppenprozessors ermittelt. Folgende Lastzustände - normal, high, overload und extreme - des Gruppenprozessors GPn werden entsprechend der Belastung der zentralen Prozessoreinheit CPU des Gruppenprozessors vom Load-Manager GPLM ermittelt. Die einzelnen Lastzustände GPLS des Gruppenprozessors GPx sind dabei beispielsweise wie folgt charakterisiert:

45
GPLS-normal:
GPLS-high :

Die Auslastung des Gruppenprozessors ist unterhalb seiner für ihn definierten Normallastschwelle. Die Normallastschwelle liegt beispielsweise bei 0,71 Erlang. (Erlang ist die Einheit des Verkehrswertes, wobei 50 0,71 Erlang einer Auslastung des Gruppenprozessors von 71 Prozent entspricht)

Der Gruppenprozessor befindet sich in Hochlast, das heißt oberhalb seiner Normallastschwelle. In 55 diesem Zustand werden noch keine Überlastabwehrmaßnahmen für den Gruppenprozessor eingeleitet.

GPLS-overload:

Der Gruppenprozessor befindet sich in Überlast. Aufgaben die vom Gruppenprozessor zu erledigen wären, werden soweit möglich, kurzfristig auf andere, nicht überlastete Gruppenprozessoren GPn übertragen. Die Überlastschwelle liegt beispielsweise bei > 0,8 Erlang.

Der auf dem Gruppenprozessor GPn implementierte Load Manager GPLM ist neben der Ermittlung der Lastzustände GPLS auch für die Bestimmung der vermittlungstechnischen Überlastabwehrstufen OPL während der Auslastphase -overload- des Gruppenprozessors GPn zuständig. Die vermittlungstechnischen Überlaststufen OPLn sind ein Maß für die Intensität der Überlastabwehr, die während sich der Gruppenprozessor GPn in Überlast befindet, von den vermittlungstechnischen Prozessoren und der dazugehörigen Peripherie durchgeführt werden können. Die vermittlungstechnischen Überlaststufen OPL werden dabei vorzugsweise in Stufen zwischen 0 und 6 unterteilt:

- OPL 0 bedeutet keine Abwehr, bei
- OPL 1 es werden ca. 16 Prozent aller neu ankommenden Calls abgewiesen, bei
- OPL 6 werden nur noch Calls akzeptiert die über priorisierte Verbindungsleitungen von einem anderen Kommunikationssystem eintreffen.

Die hier aufgeführten Lastzustände GPLS des Gruppenprozessors GPn sowie die vermittlungstechnischen Überlaststufen OPLn können beliebig, entsprechend der Dimensionierung des Kommunikationssystems K, verfeinert oder abgerundet werden.

Ist ein Gruppenprozessor GPi nicht im Lastzustand "overload", wird kein Call weiterverteilt. Ist ein Gruppenprozessor jedoch in dem Lastzustand "overload" so wird ein Anteil der neu ankommenden Calls anhand eines Wertes q, der zwischen 0 und 1 schwankt, bestimmt der zur weiteren Abarbeitung an freie Gruppenprozessoren GPx weitergeleitet werden sollte. Durch einen weiteren ermittelbaren Wert p, der die Aufnahmefähigkeit für weitere Calls der übrigen Gruppenprozessoren wiedergibt, wird festgelegt, ob aufgrund der Gesamtauslastung dieser Gruppenprozessoren GPn des Kommunikationssystems K überhaupt verteilt werden kann. Wenn entschieden ist, daß ein Call verteilt werden soll, wird entsprechend der ermittelten Werte p der Gruppenprozessoren bestimmt, welche "Ziel Gruppenprozessoren" GPx zur Abarbeitung der ECPS Prozeduren bereitgestellt werden.

Die Werte q und p werden von jedem Gruppenprozessor GPn aus den Lastzuständen GPLS und seines eigenen Lastwertes a lokal berechnet. Nachfolgend ist ein Pseudocode zur Berechnung der Werte p und q angegeben.

```
PROCEDURE calc q;
  ! calculate q;
BEGIN
  IF q=0
    THEN q:=min (1,freeGPs/overloaded GPs-0.1));
  40
```

45

50

55

```

1 IF myOLC.GPLS<2
2   THEN BEGIN ! GP not overloaded;
3     IF q<freeGPs/(overloaded:GPs+0.01)      ! q too small;
4       THEN q:=MIN(1,free GPs/(overloaded GPs+0.01))
5     ELSE q:=MIN(q*p sum,1); !no distr. ness., slowly
6       increase q
7
8   END
9
10  ELSE BEGIN !GP is overloaded;
11    IF p sum > 1      !too much distributed;
12    THEN BEGIN
13      q:=q*p sum;
14      damping:=damping+1;
15    END
16
17  ELSE BEGIN !more distribution possible;
18    IF (myOLC.OfferedLoad>0,95 OR myOLC.OPL>0) AND
19      damping>2
20    THEN BEGIN
21      q:=MIN(q*1.2,1);
22      damping:=0;
23    END
24
25  ELSE damping:=damping+1;
26  IF myOLC.offeredLoad<0.65 AND myOLC.OPL=0
27  THEN BEGIN
28    q:=MAX(0,q/1.1);
29    damping:=damping+1;
30    END;
31
32  END;
33
34  END GP is overloaded;
35
36  END PROCEDURE calc q;
37
38  END CLASS smartGPL TAB;
39
40
41
42
43
44
45  PROCEDURE calc p;
46    ! calculate p(i);
47
48  BEGIN
49    INTEGER i;
50    REAL alpha; !Initializationprob. for p;
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
287
288
289
289
290
291
292
293
294
295
296
297
297
298
299
299
300
301
302
303
304
305
306
307
308
309
309
310
311
312
313
314
315
316
317
318
319
319
320
321
322
323
324
325
326
327
328
329
329
330
331
332
333
334
335
336
337
338
339
339
340
341
342
343
344
345
346
347
348
349
349
350
351
352
353
354
355
356
357
358
359
359
360
361
362
363
364
365
366
367
368
369
369
370
371
372
373
374
375
376
377
378
379
379
380
381
382
383
384
385
386
387
388
389
389
390
391
392
393
394
395
396
397
398
399
399
400
401
402
403
404
405
406
407
408
409
409
410
411
412
413
414
415
416
417
418
419
419
420
421
422
423
424
425
426
427
428
429
429
430
431
432
433
434
435
436
437
438
439
439
440
441
442
443
444
445
446
447
448
449
449
450
451
452
453
454
455
456
457
458
459
459
460
461
462
463
464
465
466
467
468
469
469
470
471
472
473
474
475
476
477
478
479
479
480
481
482
483
484
485
486
487
488
489
489
490
491
492
493
494
495
496
497
498
499
499
500
501
502
503
504
505
506
507
508
509
509
510
511
512
513
514
515
516
517
518
519
519
520
521
522
523
524
525
526
527
528
529
529
530
531
532
533
534
535
536
537
538
539
539
540
541
542
543
544
545
546
547
548
549
549
550
551
552
553
554
555
556
557
558
559
559
560
561
562
563
564
565
566
567
568
569
569
570
571
572
573
574
575
576
577
578
579
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
599
600
601
602
603
604
605
606
607
608
609
609
610
611
612
613
614
615
616
617
618
619
619
620
621
622
623
624
625
626
627
628
629
629
630
631
632
633
634
635
636
637
638
639
639
640
641
642
643
644
645
646
647
648
649
649
650
651
652
653
654
655
656
657
658
659
659
660
661
662
663
664
665
666
667
668
669
669
670
671
672
673
674
675
676
677
678
679
679
680
681
682
683
684
685
686
687
687
688
689
689
690
691
692
693
694
695
696
697
697
698
699
699
700
701
702
703
704
705
706
707
708
709
709
710
711
712
713
714
715
716
717
718
719
719
720
721
722
723
724
725
726
727
728
729
729
730
731
732
733
734
735
736
737
738
739
739
740
741
742
743
744
745
746
747
748
749
749
750
751
752
753
754
755
756
757
758
759
759
760
761
762
763
764
765
766
767
768
769
769
770
771
772
773
774
775
776
777
778
779
779
780
781
782
783
784
785
786
787
787
788
789
789
790
791
792
793
794
795
796
797
797
798
799
799
800
801
802
803
804
805
806
807
808
809
809
810
811
812
813
814
815
816
817
817
818
819
819
820
821
822
823
824
825
826
827
828
829
829
830
831
832
833
834
835
836
837
838
838
839
839
840
841
842
843
844
845
846
847
848
848
849
849
850
851
852
853
854
855
856
857
858
859
859
860
861
862
863
864
865
866
867
868
868
869
869
870
871
872
873
874
875
876
877
877
878
878
879
879
880
881
882
883
884
885
886
887
887
888
888
889
889
890
891
892
893
894
895
895
896
896
897
897
898
898
899
899
900
901
902
903
904
905
905
906
906
907
907
908
908
909
909
910
911
912
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496
1497
1497
1498
1498
1499
1499
1500
1500
1501
1501
1502
1502
1503
1503
1504
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1510
1511
1511
1512
1512
1513
1513
1514
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1520
1521
1521
1522
1522
1523
1523
1524
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1530
1531
1531
1532
1532
1533
1533
1534
1534
1535
1535
1536
1536
1537
1537
1538
1538
1539
1539
1540
1540
1541
1541
1542
1542
1543
1543
1544
1544
1545
1545
1546
1546
1547
1547
1548
1548
1549
1549
1550
1550
1551
1551
1552
1552
1553
1553
1554
1554
1555
1555
1556
1556
1557
1557
1558
1558
1559
1559
1560
1560
1561
1561
1562
1562
1563
1563
1564
1564
1565
1565
1566
1566
1567
1567
1568
1568
1569
1569
1570
1570
1571
1571
1572
1572
1573
1573
1574
1574
1575
1575
1576
1576
1577
1577
1578
1578
1579
1579
1580
1580
1581
1581
1582
1582
1583
1583
1584
1584
1585
1585
1586
1586
1587
1587
1588
1588
1589
1589
1590
1590
1591
1591
1592
1592
1593
1593
1594
1594
1595
1595
1596
1596
1597
1597
1598
1598
1599
1599
1600
1600
1601
1601
1602
1602
1603
1603
1604
1604
1605
1605
1606
1606
1607
1607
1608
1608
1609
1609
1610
1610
1611
1611
1612
1612
1613
1613
1614
1614
1615
1615
1616
1616
1617
1617
1618
1618
1619
1619
1620
1620
1621
1621
1622
1622
1623
1623
1624
1624
1625
1625
1626
1626
1627
1627
1628
1628
1629
1629
1630
1630
1631
1631
1632
1632
1633
1633
1634
1634
1635
1635
1636
1636
1637
1637
1638
1638
1639
1639
1640
1640
1641
1641
1642
1642
1643
1643
1644
1644
1645
16
```

```

alpha:=1/(4*(freeGPs+0.1));
p_sum:=0;
FOR i:=1 STEP 1 UNTIL NumberOfGPs DO BEGIN
 5   IF the GPL TAB(i)=0
    THEN BEGIN
      IF p(i)=0 THEN p(i):=alpha ELSE p(i):=p(i)*1.05;
10
    END;
    IF the GPL TAB(i)=1
    THEN p(i):=max(0,p(i)*0.9);
15
    IF the GPL TAB(i)>1
    THEN p(i):=0;
    p sum:=p sum+p(i);
    END;
20
    IF p sum>0 THEN BEGIN !normalize p();
      FOR j:=1 STEP 1 UNTIL NumberOfGPs DO
        p(j):=p(j)/p sum;
25
    END
  END PROCEDURE calc p;

```

In Fig. 4 ist ein Ablaufdiagramm, in dem die einzelnen Programmprozeduren angedeutet sind, die zur 30 Abarbeitung von beispielsweise lokal auftretenden Überlasten an einen Gruppenprozessor GPj aufgerufen und abgearbeitet werden.

Die Berechnung von den im Verteilschema Fig. 3 angegebenen Werten $p_i(n)$, q_j sowie der Lastzustand GPLSj geschieht wie oben bereits erwähnt im Programmodul Load Manager GPLM des Gruppenprozessors GPj. Der GPLM ist ein Prozeß der von der virtuellen Prozessoreinheit VCPU-OAMH abgearbeitet wird.

Nachfolgend werden die im Ablaufdiagramm aufgezeigten Schritte beschrieben:

- Berechnung des Lastwertes a:
Der Lastwert a wird aus der Belastung der zentralen Prozessoreinheit des jeweiligen Gruppenprozessors ermittelt.

- Bestimmung des Load States GPLS eines Gruppenprozessors GPs:

Zur Bestimmung des Load State GPLS wird ein integrierender Startindikator S, der in der Art eines "Schätzers" eine andauernde Überlastsituation erkennt, ein Load Balancing Flag LBF sowie x eine Hysteresebreite, der beispielsweise der Wert 0,1 zugewiesen werden kann, verwendet.

Die Bestimmung der quantifizierten Größen GPLS wird nach folgendem Pseudocode durchgeführt:
a) Bisheriger GPLS "Normal"

```

45
      if a > Normallastschwelle + x
      then GP LS := "high";
      if S > Thresh and a > Hochlastschwelle + x
50      then GPLS := "overload";

```

b) Bisheriger GPLS "high"

```

        if a < Normallastschwelle
        then GPLS := "normal"
5       if S > Thresh und a > Hochlastschwelle + x
        then GPLS := "overload";

```

c) Bisheriger GPLS "overload"

```

10      if LBF= true
        then a':= a/(1- q(j)d)
        else a':= a;
15      if a' > Hochlastschwelle or OPL > 0
        then begin
                GPLS:="overlast";
20          real overlact:= true;
          if (OPL = 6 since 3s)
            Then GPLS := "extrem"
25      end
      esle begin   (Ab hier liegt keine echte Überlast mehr vor)


```

```

30      if no rejection since 6s
        then begin
                if a`< Hochlastschwelle
35      then begin
                GPLS := "high"
                if a`< normallastschwelle
                    then GPLS := "normal"
40      end;
                end
        else begin
                GPLS := "overload";
                real overload:= false;
45      end;
        end;
50      end;

```

d) Bisheriger GPLS = "extrem"

Von den eigenen Anschlußeinheiten LTGs wird kein neuer Lastverkehr mehr angenommen.

55 Die Programmprozedur "Bestimmung GPLS" wird im wesentlichen aus dem Lastwert a des Prozessors der letzten Sekunde bestimmt. Falls bereits verteilt wird, wird über die Größe a', die die tatsächliche an den Gruppenprozessor anliegende Last ausdrückt, berechnet. Der Lastwert a' stellt einen Schätzer für die zu erwartende Last für die zentrale Prozessoreinheit CPU des nächsten Kontrollintervalls dar.

- Informieren der Load Manager auf allen GPn über die Lastzustände aller GPn
Zwei Möglichkeiten der Informationsverteilung kommen dazu in Betracht:

- a) Alle GPLMs senden ihre lokale Lastinformation (GPLS, OPL) an einen zentralisierten Prozeß, Load Control Master (LOCM) genannt, auf einem ganz bestimmten Prozessor GPx. Der LOCM sammelt die Lastinformationen und sendet so rasch als möglich die gesammelten Informationen an alle GPLMs im System.
- b) Mittels einer 'jeder-informiert-jeden' Kommunikation zwischen den GPLMs verteilt jeder GPLM seine lokale Lastinformation (GPLS, OPL) an die GPLM aller anderen GPn im System.

Nach einem kurzen Zeitintervall der Zwischenspeicherung aller GPLS werden diese an alle GPn verteilt und dort jeweils in eine Tabellen eingetragen. Nach einem kurzem Zeitintervall, stehen aktualisierte GPLS von jedem Gruppenprozessor GPx zur Eintragung in eine Gruppenprozessortabelle bereit.

- Bestimmung der zusätzlich abzuarbeitenden Aufgaben p(i):

Der Wert $p(i)$ ist dabei der Anteil der von den "Heim- Gruppenprozessoren" wegverteilten Calls C, die auf einen freien Gruppenprozessor GP(j) bearbeitet werden sollen. Der Wert $p(i)$ wird dabei nach folgender Strategie bestimmt:

- a) GPLS(i) = "normal"

$p(i)$ kann erhöht werden, da auf GP(i) noch Bearbeitungskapazität ausgenutzt werden kann.

- b) GPLS(i) = "high"

$p(i)$ wird verringert, da GP(i) zuviel Last hat.

- c) GPLS(i) = "overload"

$p(i)$ wird auf Null gesetzt, da GP(i) in einem Betriebszustand der Überlast ist.

Bei der Neueinstellung der Werte $p(i)$ wird von einem durchschnittlichen Verteilanteil pro freien Gruppenprozessor GP(i) ausgegangen. Solange sich ein GP(i) im Lastzustand "overload" befindet, ist $p(i)=0$. Wechselt der Lastzustand von "overload" in "normal" oder "high" so wird der Wert $p(i)=1/(C \cdot \text{Anzahl der freien GP}(x))$ festgelegt. C ist ein Parameter, in dem vermittelungsspezifische Überlegungen betreffend der Vermeidung eines Überschwingens des Lastverteilungssystems in der Einschwingphase eingeht.

- Bestimmung der wegzuteilenden Aufgaben q(j)

Bevor der Wert $q(j)$, der den Anteil der wegzuteilenden Calls angibt, wenn ein Gruppenprozessor im Lastzustand Überlast ist, wird an Hand der $p(i)$ festgestellt ob überhaupt verteilt werden kann. Ergibt die Summe der noch nicht normierten $p(n)$ einen Wert der größer als 1 ist, so wird der Wert $q(j)$ nach den Erfordernissen des jeweiligen GP(j) bestimmt. Anhand des Lastwertes a' wird festgestellt, ob $q(j)$ erhöht oder erniedrigt werden sollte. Falls der GP(j) nicht im Lastzustand "overload" ist, wird immer versucht $q(j)$ zu erhöhen.

- a) Falls die vermittelungstechnische Überlaststufe OPL > 0 oder Lastwert $a > 0,8 + x$ (mit $x = 0,1$) ist, wird $q(j)$ erhöht.

- b) Falls GPLS(j) = "overload" und OPL = 0 und Lastwert $a < 0,8 - x$ ist wird $q(j)$ verringert.

c)

40 Falls GPLS(i) < "overload" erhöhe q(i)

Die Bestimmung des Wertes $q(j)$ im Fall c bezieht sich auf den Einstieg in die Lastverteilung.

Normalerweise sollte dann $q(j) = 1$ sein, um bei plötzlicher Überlast maximale Verteilung zu bekommen.

Wenn das Kommunikationssystem K an seiner Leistungsgrenze arbeitet, muß $q(j)$ kleiner sein, um die Lastverteilung, durch Eintritt zusätzlicher Gruppenprozessoren GPx in Überlast nicht zu gefährden. Aus diesem Grund wird $q(j)$ verringert, falls die Summe der einzelnen $p(n)$ kleiner als 1 ist.

Um mögliche Schwingungen des Überlastverfahrens zu vermeiden, ist es vorteilhaft, wenn vor einer Verringerung des Wertes $q(j)$ ein kurzer Zeitintervall abgewartet wird. Nachdem festgestellt wurde, daß der Gruppenprozessor GP(j) Calls verteilen möchte, wird das Load Balancing Flag LBF "true", wenn nicht, wird das LBF "false" gesetzt. In einem mit Guard Period bezeichneten Schutzintervall, wird das Load Balancing Flag LBF := false gesetzt, obgleich der Gruppenprozessor GP(j) sich im Lastzustand "overload" befinden kann.

Bei einer Neueinstellung des Wertes $q(j)$ wird folgender Algorithmus angewendet:

Falls $q(j) = 0$ wird $q(j) := \min(1, (\text{Anzahl freier GPx} / \text{Anzahl der GPs die im Lastzustand "overload" sind}))$.

- Ist der GPLS von GP(j) nicht im "overload" und $q(j) < (\text{Anzahl freier GPs} / \text{Anzahl der GPs in "overload"})$, so wird $q(j) := \min(1, \text{Anzahl freier GPs} / \text{Anzahl der GPs in "overload"})$ gesetzt. Wenn der GP(j) nicht in "overload" ist, ist normalerweise $q(j) = 1$, um einen schnellen Einstieg in die Lastverteilung zu gewährleisten, falls der GP(j) plötzlich in Überlast gerät. Falls jedoch mehr GPs in "overload" sind als GPs zur Aufnahme von

fremder Überlast bereitstehen, wird $q(j)$ passend kleiner gewählt, um die nicht überlasteten GPs vor Überlast zu schützen und damit das System der Lastverteilung nicht zu gefährden.

Das Verteilen der Calls erfolgt wie nachfolgend angegeben: Beim Anliegen eines Calls, stellt der Service Manager SM, der eine direkte Schnittstelle zu den zu bearbeitenden Calls darstellt und die eigentliche Verteilung der Calls vornimmt anhand des LBF fest, ob verteilt werden soll oder nicht. Falls verteilt werden soll, wird untersucht, ob es möglich ist den anliegen Call zu verteilen. Falls die Verteilung im Prinzip möglich und erwünscht ist, wird entschieden, ob dieser spezielle Call verteilt werden soll. Dazu wird der Quotient aus den bisher in dem Zeitintervall verteilten Calls mit der Anzahl aller ankommenden Calls gebildet und mit $q(j)$ verglichen. Ist der Quotient größer als $q(j)$, wird der Call verteilt. Ist beispielsweise die Anzahl der bisher angekommenen Calls 20, die der bisher verteilten Calls 4 und der Wert $q(j)=0,25$, so ergibt sich aus dem Quotienten 4/20, daß dieser kleiner als $q(j)$ ist. In diesem Fall wird der Call verteilt. Wenn der Call verteilt werden soll, wird dazu analog anhand der Werte von $p(j)$ entschieden, welcher Gruppenprozessor GPx die Bearbeitung des Calls übernimmt.

15 Patentansprüche

1. Verfahren zum Lastenausgleich in einem Multiprozessorsystem, insbesondere ein Multiprozessorsystem eines Kommunikationssystems (K), bei dem anfallende Aufgaben von mehreren Prozessoren (GPn) unter Realzeitbedingungen abgearbeitet werden können,
mit folgenden Verfahrensschritten:
 - jeder Prozessor(GPi) ermittelt seinen Lastzustand(GPLSi) in Form einer quantifizierten Größe,
 - jedem Prozessor(GPi) werden die Lastzustände(GPLSn) der anderen Prozessoren(GPn) innerhalb eines Zeitrasters mitgeteilt,
 - jeder Prozessor(GPi) gibt in Abhängigkeit vom Überschreiten einer bestimmten Größe seines Lastzustandes(GPLSi) und in Abhängigkeit von den Lastzuständen der übrigen Prozessoren(GPn) zumindest einen Teil der bei ihm anfallenden Aufgaben an die übrigen Prozessoren(GPn) ab,
 - die abgegebenen Aufgaben werden entsprechend der Lastzustände (GPLSn) der übrigen Prozessoren(GPn) auf diese aufgeteilt.
2. Verfahren nach Anspruch 1,
dadurch gekennzeichnet,
daß ein Prozessor(GPi) nur so viele Aufgaben abgibt, daß sein Lastzustand(GPLS) wieder die genannte bestimmte Größe unterschreitet.
3. Verfahren nach Anspruch 1,
dadurch gekennzeichnet,
daß die die Lastzustände charakterisierenden quantifizierten Größen jeweils in deren oberen sowie unteren Bereichsübergängen mit einem Hysteresewert versehen werden.
4. Verfahren nach Anspruch 1,
dadurch gekennzeichnet,
daß bei Abgabe von Aufgaben von einem Prozessor(GPi) an die übrigen Prozessoren (GPn) zusätzlich ein Maß für eine Aufgabenabwehr des Prozessors(GPi) bestimmt wird.
5. Verfahren nach Anspruch 1,
dadurch gekennzeichnet,
daß die prozessorbezogenen Lastzustände (GPLSn) mittels einer 'jeder-informiert-jeden' Kommunikation von den einzelnen Control Managern der Prozessoren (GPn) direkt an alle anderen Control Managern der Prozessoren (GPn) im Kommunikationssystem (K) verteilt werden.
6. Verfahren nach Anspruch 1,
dadurch gekennzeichnet,
daß die prozessorbezogenen Lastzustände(GPLSn) im Load Control Manager (LOCM) eines bestimmten Prozessors(GPx) zusammengefaßt werden.
7. Verfahren nach Anspruch 6,
dadurch gekennzeichnet,

EP 0 645 702 A1

daß entsprechend eines Zeitrasters die in dem bestimmten Prozessor(GPx) zusammengefaßten Lastzustände(GPLSn) jeweils in Tabellen der Prozessoren(GPn) abgespeichert werden.

8. Verfahren nach einem der vorhergehenden Ansprüche,
dadurch gekennzeichnet,
daß in Abhängigkeit vom jeweiligen Lastzustand (GPLS) der Prozessoren der Anteil ($p_i(n)$) der zusätzlich durch sie noch abarbeitbaren Aufgaben festgelegt wird.
9. Verfahren nach einem der vorherigen Ansprüche,
dadurch gekennzeichnet,
daß in Abhängigkeit vom jeweiligen Lastzustand (GPLSn) der Prozessoren (GPi) die von diesem wegzuteilenden Aufgaben ($q(j)$) erhöht oder vermindert werden.
10. Verfahren nach einem der vorherigen Ansprüche,
dadurch gekennzeichnet,
daß in jedem Prozessor in Abhängigkeit vom Summenergebnis, aus den Anteilen ($p(i)$) der durch die übrigen Prozessoren zusätzlich zu übernehmenden Aufgaben, der Anteil der wegzuteilenden Aufgaben ($q(j)$) im Überlastfall bestimmt wird.

20

25

30

35

40

45

50

55

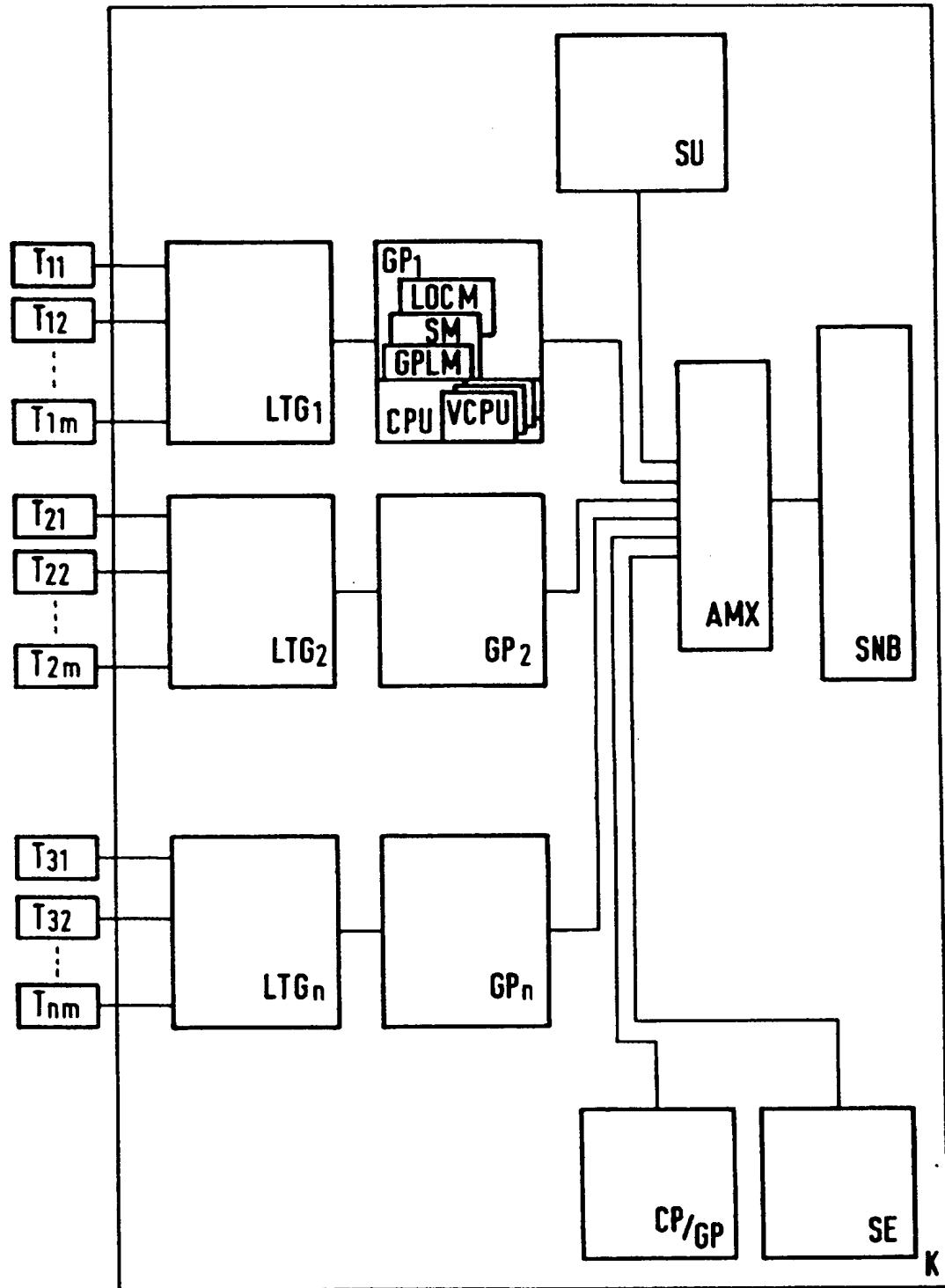


FIG 1

FIG 2

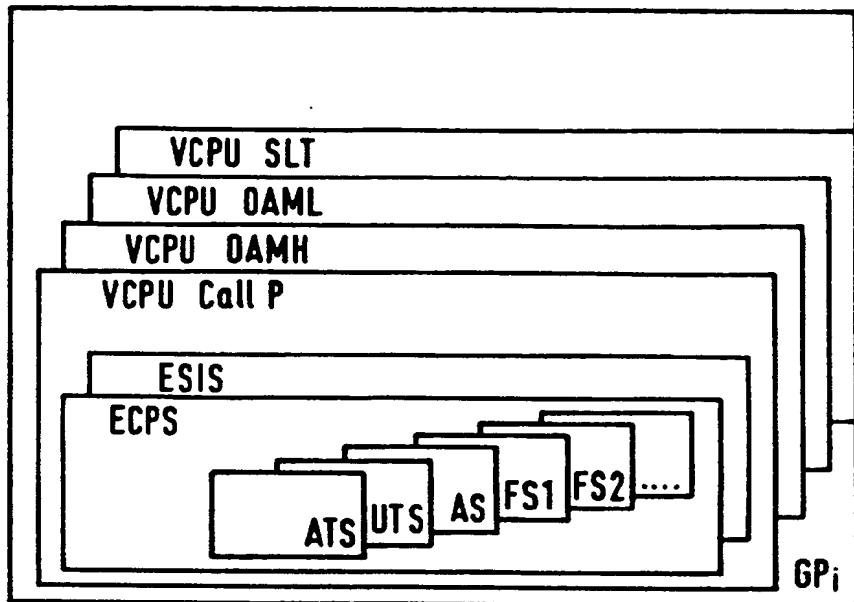


FIG 3

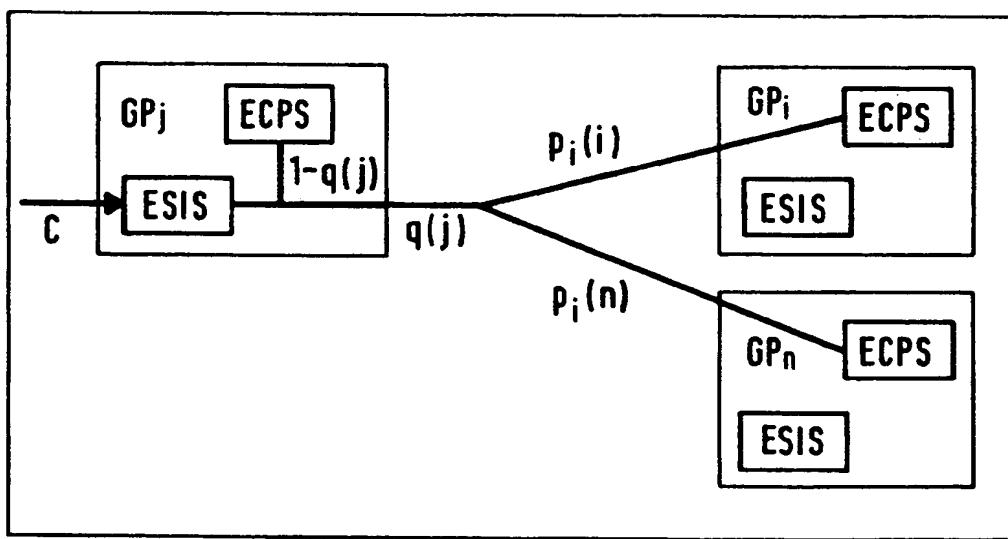
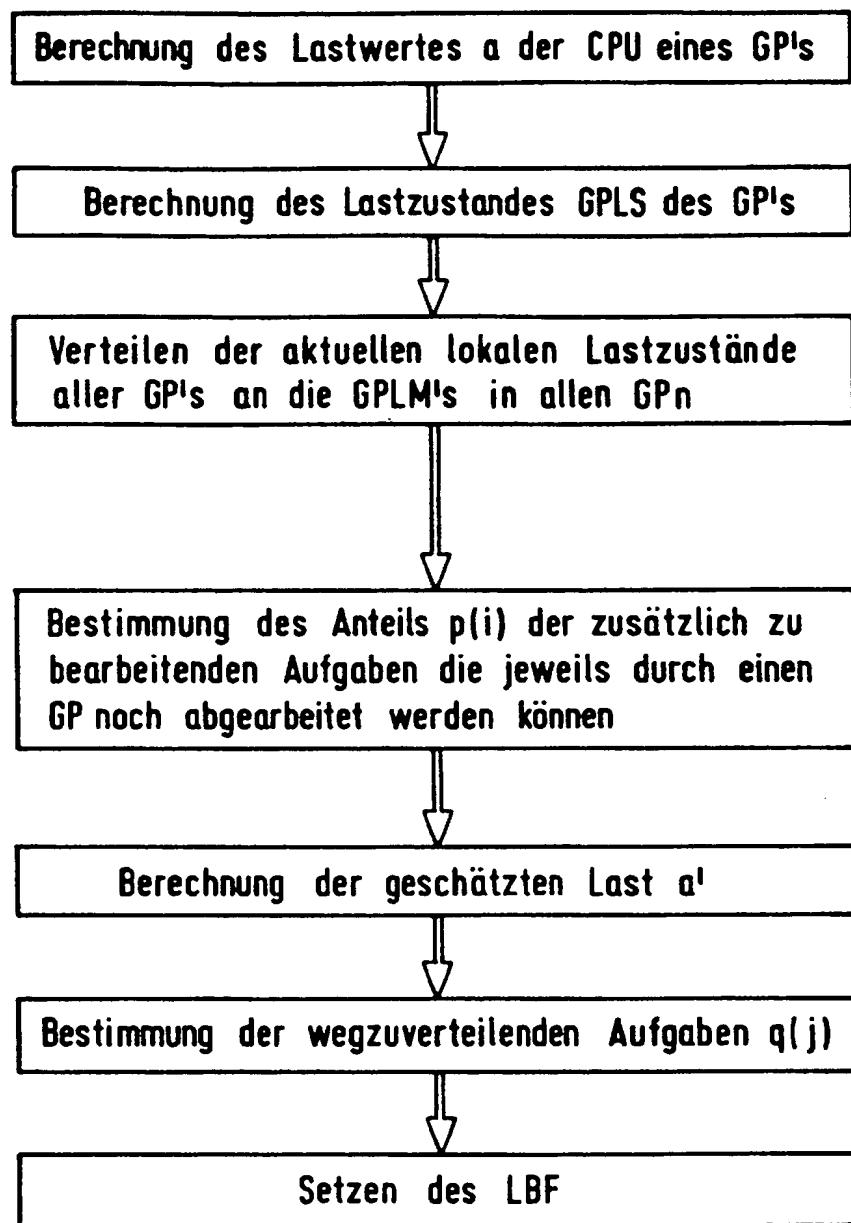


FIG 4





Europäisches
Patentamt

EUROPÄISCHER RECHERCHENBERICHT

Nummer der Anmeldung
EP 93 11 5467

EINSCHLÄGIGE DOKUMENTE			
Kategorie	Kennzeichnung des Dokuments mit Angabe, soweit erforderlich, der maßgeblichen Teile	Betreff Anspruch	KLASSIFIKATION DER ANMELDUNG (Int.Cl.)
X	SOFTWARE PRACTICE & EXPERIENCE Bd. 19, Nr. 5, Mai 1989, CHICHESTER GB Seiten 411 - 435 XP98633 PIYUSH DIKSHIT ET AL. 'SAHAYOG: a test bed for evaluating dynamic load-sharing policies' * Seite 419, Zeile 6 - Zeile 19 * * Seite 425, Zeile 21 - Seite 426, Zeile 10 *	1,2,5-7, 9	G06F9/46 H04M3/36 H04Q3/545
Y	---	3,4,8,10	
Y	IEEE TRANSACTIONS ON COMPUTERS Bd. 38, Nr. 8, August 1989, NEW YORK US Seiten 1124 - 1142 XP47581 KANG G. SHIN ET AL. 'Load sharing in distributed real-time systems with state-change broadcasts' * Seite 1124, linke Spalte, Zeile 1 - Seite 1125, rechte Spalte, Zeile 10 * * Seite 1125, rechte Spalte, Zeile 32 - Seite 1126, linke Spalte, Zeile 15 * * Seite 1127, linke Spalte, Zeile 16 - Zeile 58 *	3,4,8,10	RECHERCHIERTE SACHGEBIETE (Int.Cl.)
A	PROC. 12TH INT. CONF. ON DISTRIBUTED COMPUTING SYSTEMS 9. Juni 1992, IEEE, LOS ALAMITOS, US Seiten 82 - 89 XP341000 JUNGUK KIM ET AL. 'An all-sharing load balancing protocol in distributed systems on the CSMA/CD local area network' * Seite 83, linke Spalte, Zeile 52 - rechte Spalte, Zeile 16 * * Seite 84, linke Spalte, Zeile 35 - rechte Spalte, Zeile 29 * ---	10 -/-	G06F H04M H04Q
Der vorliegende Recherchenbericht wurde für alle Patentansprüche erstellt			
Recherchenort	Abschlußdatum der Recherche		Prüfer
DEN HAAG	26. November 1993		Kingma, Y
KATEGORIE DER GENANNTEN DOKUMENTE			
X : von besonderer Bedeutung allein betrachtet	T : der Erfindung zugrunde liegende Theorien oder Grundsätze		
Y : von besonderer Bedeutung in Verbindung mit einer anderen Veröffentlichung derselben Kategorie	E : älteres Patentdokument, das jedoch erst am oder nach dem Anmeldedatum veröffentlicht worden ist		
A : technologischer Hintergrund	D : in der Anmeldung angeführtes Dokument		
O : nichtschriftliche Offenbarung	L : aus anderem Gründen angeführtes Dokument		
P : Zwischenliteratur	& : Mitglied der gleichen Patentfamilie, übereinstimmendes Dokument		



Europäisches
Patentamt

EUROPÄISCHER RECHERCHENBERICHT

Nummer der Anmeldung
EP 93 11 5467

EINSCHLÄGIGE DOKUMENTE			
Kategorie	Kennzeichnung des Dokuments mit Angabe, soweit erforderlich, der maßgeblichen Teile	Betreff Anspruch	KLASSIFIKATION DER ANMELDUNG (Int.Cl.6)
A	PROC. 11TH INT. TELETRAFFIC CONGRESS 4. September 1985 , ELSEVIER, AMSTERDAM, NETHERLANDS Seiten 835 - 841 M. VILLENA ALTAMIRANO ET AL. 'An overload control strategy for distributed control systems' * das ganze Dokument * ---	2,8	
A	US-A-4 974 256 (BERNARD L. CYR ET AL.) * Spalte 2, Zeile 5 - Spalte 3, Zeile 13 * -----	4,8-10	
RECHERCHIERTE SACHGEBIETE (Int.Cl.6)			
Der vorliegende Recherchenbericht wurde für alle Patentansprüche erstellt			
Recherchenort	Abschlußdatum der Recherche	Prüfer	
DEN HAAG	26. November 1993	Kingma, Y	
KATEGORIE DER GENANNTEN DOKUMENTE			
X : von besonderer Bedeutung allein betrachtet	T : der Erfindung zugrunde liegende Theorien oder Grundsätze		
Y : von besonderer Bedeutung in Verbindung mit einer anderen Veröffentlichung derselben Kategorie	E : älteres Patentdokument, das jedoch erst am oder nach dem Anmeldeatum veröffentlicht worden ist		
A : technologischer Hintergrund	D : in der Anmeldung angeführtes Dokument		
O : nichtschriftliche Offenbarung	L : aus andern Gründen angeführtes Dokument		
P : Zwischenliteratur	& : Mitglied der gleichen Patentfamilie, übereinstimmendes Dokument		

